[MongoDB主从复制+集群](http://www.cnblogs.com/jiyukai/p/6984775.html)

**一、读写分离的概念**

读写分离，基本的原理是让主数据库处理事务性增、改、删操作（INSERT、UPDATE、DELETE），而从数据库处理SELECT查询操作。数据库复制被用来把事务性操作导致的变更同步到集群中的从数据库。

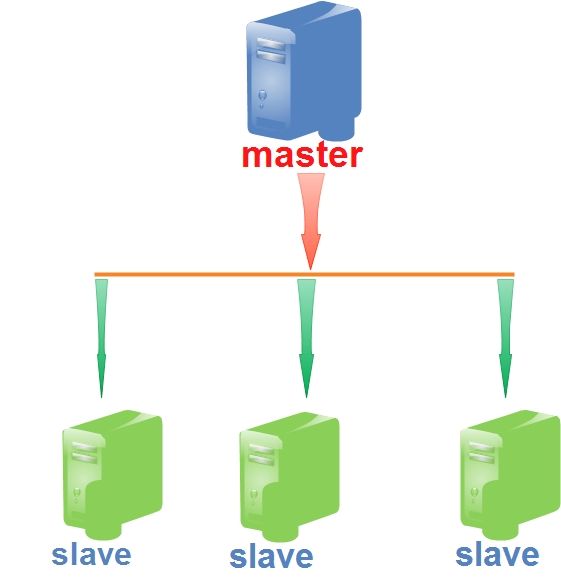
读写分离的目的是为了实现高并发场景下的请求分流，避免对数据库的访问过于集中，导致性能下降甚至是宕机。

**二、主从复制介绍**

在MongoDB的集群中，会有指定为master的主节点存在，该节点用于被客户端进行数据的增删改操作。同时集群中还会有被指定为slave的节点存在，即从节点，该节点主要接收来自于客户端的读，检索操作，并不具备增删改操作的功能。

在master节点处理完增删改操作后，会实时同步数据到与其绑定的从节点上，实现主从复制至少要有两个数据库实例，并且每个从节点需要知道主节点的地址，如果是在linux中记得开通主从端口的防火墙。

主从复制的优势在于比较灵活，适用于数据备份，故障后人工进行数据恢复，以及对读数据的扩展等。

****

**三、主从复制局限**

（1）MongoDB目前建议使用副本集实现集群管理，不建议使用简单的主从复制

（2）主从复制在master宕机后，没有自动选举master机制，导致主节点服务一挂，便不能对外提供增删改操作。

（3）所有增删改操作都是针对主节点进行操作，可能导致主节点性能下降。

（4）从节点对主节点的数据都是全量拷贝，对主从节点的压力都是不小的。

**四、主从复制简单配置及实现**

设备有限，为方便简单的实现，可在一台机器上模拟主节点和从节点。在C盘创建master目录，在E盘创建slave目录，master目录作为主节点数据存放的目录，slave目录作为从节点数据存放的目录。

注意：主节点和从节点要指定不同的端口。  
 启动主节点：mongod --dbpath C:\master --port 666 --master  
 启动从节点：mongod --dbpath E:\slave --port 888 --slave --source localhost:666

启动成功后就可以连接主节点进行操作了，而对主节点的操作会同步到从节点，而对从节点进行插入操作时，会报not master的提示并拒绝写入数据。

**配置实现解释**

master：默认为false，若要设置当前节点为主节点，需要在服务端启动添加--master  
 slave：默认为false，若要设置当前节点为从节点，需要在服务端启动添加--slave  
 source：默认为空，用于从节点，指定从节点的复制来源，即主节点所在的地址，格式为：<host><:port>  
 only：默认为空，用于从节点，主动复制默认复制主节点上所有的数据库，通过设置此项指定需要复制的数据库名称  
 slavedelay：设置从节点同步主节点的延迟时间，用于从节点设置，默认为0，单位秒。  
 autoresync：默认为false，用于从节点设置。是否自动重新同步。设置为true，如果落后主节点超过10秒，会强制从节点自动重新同步。如果oplogSize太小，此设置可能有问题。如果oplog大小不足以存储主节点的变化状态和从节点的状态变化之间的差异，这种情况下强制重新同步是不必要的。当设置autoresync选项设置为false，10分钟内从节点不会进行大于1次的自动重新同步。

**什么是oplog？**

主节点的操作会被记录为oplog,存储在系统数据库local的集合oplog.$main中，这个集合中的每个文档都代表主节点的一个操作（不包括查询），从节点定期从主服务器获取oplog数据，并在本机进行执行，oplog使用的是固定集合，随着操作的逐渐增加，新的文档会逐渐覆盖旧的文档。